

## Swipeout y Alertas

Se divide el ejercicio en tres apartados: en el primero completaremos la parte gráfica creando la lista de excursiones favoritas, en el segundo actualizaremos nuestro repositorio Redux para que realice la función de borrar y en el tercero se inserta la alerta al pulsar el botón borrar.

### 1. Swipeout

Se instala la librería *React Native Swipeout* que se usará para realizar el desplazamiento hacia la derecha para que aparezca el botón borrar:

```
yarn add react-native-swipeout
```

Se crea el componente *VistaFavoritosComponent.js* y se modifica el menú para poder acceder a él.

Se modifica el código tal y como se indica en el commit 13 y además se modifica el código para generar la lista de favoritos:

```
if(this.props.favoritos.length == 0){
  return(<View></View>);
}else{
  for(var i=0; i<this.props.favoritos.length;i++){
    if(i==0){
      var excurfav= this.props.excursiones.excursiones.filter((excursion) => excursion.id === this.props.favoritos[i]);
    }
    if(this.props.favoritos.length!=1){
      excurfav= excurfav.concat(this.props.excursiones.excursiones.filter((excursion) => excursion.id === this.props.favoritos[i+1]));
    }
  }

  return (
    <SafeAreaView>
      <FlatList
        data={excurfav}
        renderItem={renderFavoritoItem}
        keyExtractor={item => item.id.toString()}
      />
    </SafeAreaView>
  );
}
```

Como se observa cuando favoritos se encuentra vacío no devuelve nada, cuando ya no está vacío se realiza un for con dos if el primero asegura que la variable *excurfav* tenga siempre el primer elemento y luego el siguiente if lo concatena con los siguientes elementos del array. Las excursiones favoritas se almacenan en la variable *excurfav* que se pasa al *FlatList*.

### 2. Actualizar favoritos en Redux

Se crea la función *borrarFavoritos()*, para esto se modifica los *ActionTypes* y el *ActionCreator* además del *reducer favoritos.js*.

```
export const BORRAR_FAVORITO = 'BORRAR_FAVORITO';
```

```
export const borrarFavorito = (excursionId) => ({  
  type: ActionTypes.BORRAR_FAVORITO,  
  payload: excursionId  
});
```

```
case ActionTypes.BORRAR_FAVORITO:  
  return state.filter(favorito => favorito !== action.payload)
```

Como se puede observar en el reducer la acción de eliminar se realiza con un filter, en un principio se intento con un splice pero para este caso no funciona correctamente por sus características, así que se sustituyó por un filter.

Se actualiza también el componente VistaFavoritos para hacer uso de la función:

```
const mapDispatchToProps = dispatch => ({  
  borrarFavorito: (excursionId) => dispatch(borrarFavorito(excursionId))  
},  
})
```

### 3. Alert

El componente *Alert* de React Native nos permite mostrar un cuadro de diálogo de alerta con el título y el mensaje especificados.

El objetivo es lanzar un cuadro de diálogo de alerta cuando el usuario decida borrar de favoritos alguna de las excursiones.

Para ello será necesario actualizar el código de nuevo:

```
const renderFavoritoItem = ({item, index}) => {  
  const createAlert = () =>  
    Alert.alert(  
      "Borrar excursión favorita?",  
      "Confirme que desea borrar la excursión: " + item.nombre,  
      [  
        {  
          text: "Cancelar",  
          onPress: () => console.log(item.nombre + ' Favorito no borrado'),  
          style: "cancel"  
        },  
        { text: "OK", onPress: () => this.props.borrarFavorito(item.id) }  
      ]  
    );  
  const rightButton = [  
    {
```

```

        text: 'Borrar',
        type: 'delete',
        onPress: showAlert

];

return (
  <Swipeout right={rightButton} autoClose={true}>
    <ListItem
      key={index}
      bottomDivider>
        <Avatar source={{uri: baseUrl + item.imagen}} />
        <ListItem.Content>
          <ListItem.Title>{item.nombre}</ListItem.Title>
          <ListItem.Subtitle>{item.descripcion}</ListItem.Subti
title>
        </ListItem.Content>
      </ListItem>
    </Swipeout>
  );
};

```

Se añade el element Alert con dos botones, el de Cancelar y el de Ok para borrar la excursión de favoritos. Si se cancela saca por consola que la excursión no ha sido borrada.